September 1, 1978 ESL-FR-834-3

COMPLEX MATERIALS HANDLING AND ASSEMBLY SYSTEMS

Final Report

June 1, 1976 to July 31, 1978

Volume III

OPTIMIZATION OF A CLOSED NETWORK OF QUEUES

by

Giovanni Secco-Suardo

The research reported herein was carried out in the Electronic Systems Laboratory with partial support extended by National Science Foundation Grant NSF/RANN APR76-12036.

Electronic Systems Laboratory
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

ABSTRACT

Closed networks of queues models have been found by several authors to be computationally simple but surprisingly accurate in predicting the steady-state behavior of flexible manufacturing systems for given parameters. Little or no attempt has been made previously, however, to exploit their mathematical structure to implement efficient production optimization algorithms. In this study, the problem of optimal flow allocation in a system is stated using a general class of closed networks of queues, the expressions for the gradient of the objective function are derived, and it is shown that as the number of parts, N, in the system grows, the problem degenerates into a simpler min-max problem which can be stated as an LP.

The results of a case study indicate that the solutions derived using the network approach are in good agreement with the solutions derived using a multicommodity flow approach. Moreover, the solutions of the asymptotic problem seem to apply also for conditions far from saturation.

ACKNOWLEDGMENTS

The author wishes to thank Prof. Businaro, Director of FIAT Research Center, for the opportunity given to him to visit MIT as a Fellow in the Advanced Study Program, Center for Advanced Engineering Study. Special thanks are due to Ing. Vincenzo Nicolo', for his keen insights and suggestions and his friendly support.

The author has greatly benefited from the patient criticism of Stanley Gershwin and the exhaustive discussions with Joseph Kimemia. Special thanks are due to Prof. Michael Athans for his original suggestion of undertaking this work.

Preface (December 1, 1979 revisions)

Professor William Maxwell of Cornell University point out an error in the case study of Section 5 in the original September 1, 1978 printing of this report. The error was corrected by Mr. Joseph Kimemia, and pages 33, 34, 35, and 40 have been revised accordingly. An errata sheet is available for holders of the original report.

CONTENTS

FIGURES

1. INTRODUCTION

This volume deals with the optimization of flows in a flexible manufacturing system. This problem, also referred to as scheduling or loading, is usually approached by considering only the workload time constraints at the machines and disregarding an important component of the total production time, namely the waiting times.

To include this aspect in the problem, one has first to define a model which describes how a given set of flows affects the queues and the overall throughput.

In Vol.II of this report [23] an approach is presented which models the system as a multicommodity flow network. This volume presents an alternative approach which models the system as a closed network of queues.

Closed networks of queues provide a satisfactory analytical model with which to study the steady state behavior of flexible manufacturing systems. These models have been used mostly as a fast and cheap alternative for more expensive simulation programs in what-if or exhaustive search analysis [1,2]. Little or no attempt has been made to exploit their mathematical structure to implement efficient optimization algorithms. This volume describes a few preliminary results in this direction.

The problem of optimal flow allocation in the system is stated using a general class of closed networks of queues, which is simple enough to be handled, but at the same time very effective as a model. In Section 2 we define this class and we state the expressions to be used by the optimization algorithm. The optimal flow problem is then stated and discussed in Section 3.

The features of the problem, when the system approaches saturation are examined in Section 4, where we show that the optimal flow allocation satisfies an LP problem. A case-study is examined in Section 5 to clarify the approach and leads to several interesting insights concerning the properties of the optimal solution. It has been found that optimal allocation is not necessarily "balanced" (i.e., equivalent

machines may have different workloads at the optimal point). Furthermore, the solution to the asymptotic problem is a good initial guess for the algorithm even at production levels not close to the saturation. Areas of future work are briefly indicated in Section 6, and the conclusions are drawn in Section 7.

2. MODEL DESCRIPTION AND ANALYSIS

2.1 Introduction

The objective of this section is two-fold: (1) to describe the class of networks which are the models assumed by our optimization approach; (2) to state in a self-contained fashion the algorithms required to analyze the model and to optimize its performance.

The class of networks discussed is substantially the same as those introduced recently by Solberg [1] to model a flexible manufacturing system. Its accuracy is amazing considering the assumptions of the model.

The original interest in this model and the first analysis of it are due to Jackson [9], who identified its potential as a tool to examine job-shop operations. The further developments of the theory must be credited to computer scientists, who used the model to evaluate computer performance [5]. Thanks to their effort good algorithms are available nowadays to analyze it.

This section is an attempt to draw together the published results relevant to the optimization problem. Some effort has been made to present them in a self-contained way, as the body of the relevant literature is widely scattered.

In Section 2.3 the steady state solution for the network state is derived following Muntz's approach [15], which is the most appealing. Buzen's algorithm [12] is derived in Section 2.4, while Section 2.5 presents the generating function approach [14], which can be considered a z-domain version of Buzen's convolutional method in the n-domain. This approach is used to derive some closed form expressions for quantities of interest.

The remaining sections, except for the last, state the best published algorithms to compute the performance quantities relevant to the optimization problem.

Section 2.10 examines the behavior of the model in conditions approaching saturation, which is probably the prevailing operation condition for a production system. The results presented in the section

show that for large N the multicommodity flow approach [see Vol. II] and our approach are consistent.

No result in this section is new. The proof of the major theorem presented in the last section appears to be new.

2.2 Model Assumptions

The optimization problem assumes a network of queues having the following properties:

(i) The network includes M stations, which can be either single-server (SS), multiple-server (MS) (in which station i has L_i servers) or infinite-server (IS);

(ii) Each station operates according to a first come-first serve (FCFS) discipline;

(iii) The network is closed i.e., N clients (referred to also as pallets or workpieces) circulate in the system;

(iv) Only one class of clients is allowed;

(v) Service times are exponentially distributed;

(vi) Transition probabilities (p_{ij}) from one station to another are time- and state-independent and define a single chain embedded Markov process;

(vii) Each station has room for N clients.

Such a class of networks has the advantage of being analytically tractable [3,4] and is capable of effectively modelling a flexible manufacturing system [1,2].

Modelling is a task which can be carried out in a relatively straightforward manner, with the possible exception of the transportation system the modelling of which requires some ingenuity [1,2]. For a special type of transportation system, a MS' station operating (i.e. the transportation system acting as a central server) like the C.P.U. of Buzen's model of a computer [5] has been suggested and has proven quite

effective [1]. This same topology has been used to model a conveyor belt type material handling subsystem also with good results [2]. The procedure to implement this model is described in another volume of this report [2].

For more complicated lay-outs (e.g., multiple loops) aggregating all transportation links into one central server may mask possible bottlenecks at some links. In these instances, it may be preferable to model each or some of the links as an MS station (Fig. 1) with a number of servers equal to the physical capacity of the link.

A MS station can also be modelled as an IS station if the probability of waiting at the station is remote. This practice is recommended because it speeds up the computation, as shown below.

Before dealing with computational issues, it is worth mentioning that this model has shown to be satisfactory indeed and this fact has puzzled the experts since many assumptions of the model are by no means realistic. Assumption (ii) and (vi) rule out any control scheme which assign stations or priorities to workpieces according to some rationale. No dispatching strategy [6,7] violating assumption (ii) can be accounted. The processing time at any station is modelled as a single exponentially distributed random variable, according assumptions (iv) and (v).

An explanation of the good behavior of the model is yet to be found [8].

2.3 Steady State Probability Density Function of the Network

The model just discussed is a special case of the closed Jackson network [9]. The state of the system at any time is defined by the vector

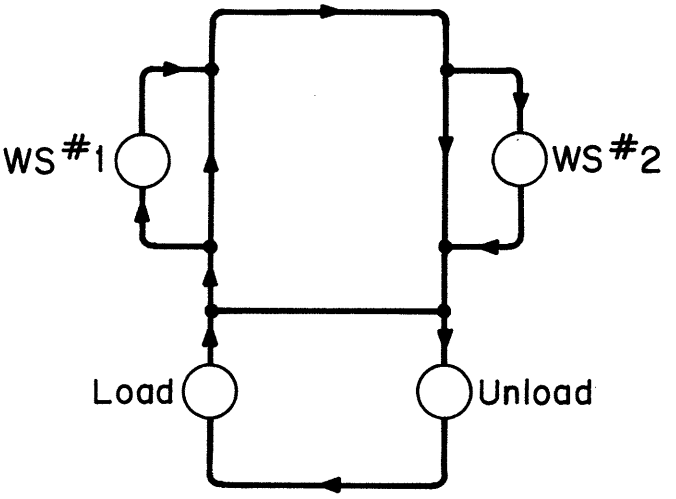
$$\underline{n} = \{n_1, n_2, \ldots, n_M\}$$

where

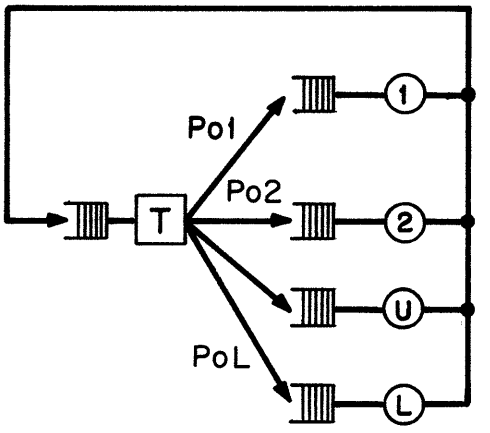
n_i = number of clients at station i (waiting or being served)

EXAMPLE

PHYSICAL LAY-OUT:



MODEL 1: CENTRALIZED TRANSPORTATION SYSTEM



MODEL 2: DISTRIBUTED TRANSPORTATION SYSTEM

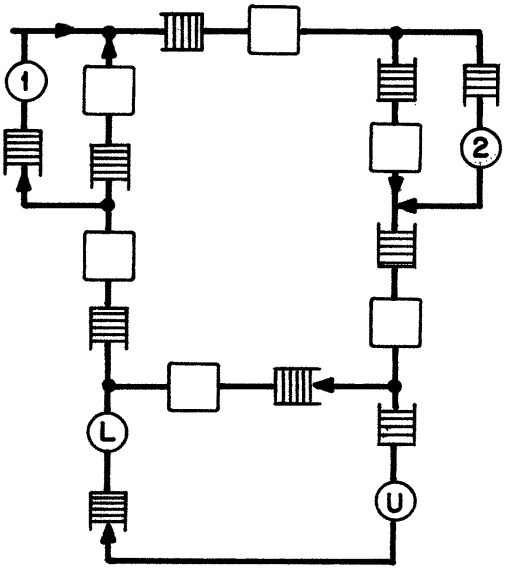


Fig. 1 Comparison of Centralized and Distributed Transportation System Models

Solutions for the steady state probability density function (p.d.f.) of $\underline{n}$ were found by Jackson [9], and by Gordon and Newell [3].

The model can also be considered as a subset of a wider class of networks studied by Basket, Chandy, Muntz and Palacios [4]. Queues included in this class are of the so called M → M type [10], which enjoy the property that if the incoming flow of clients is Poisson, the outgoing flow is also Poisson. Muntz showed that networks, which integrate M → M queues and where clients flow from one queue to the other with transition probabilities p_{ij} independent of time or state have a steady state p.d.f. of the product form [10,15]. The construction of the steady state p.d.f. $\underline{P}(\underline{n})$ requires as a preliminary step the computation of the relative flows in the network. Let

e_i = flow of clients at station i

p_{ij} = transition probability of a client from station i to station j

Then e_i must satisfy the flow balance equations:

$$e_j = \sum_{i=1}^{M} p_{ij} e_i \qquad j=1, \ldots, M \tag{1}$$

Since the system is homogeneous, it is clear that the absolute level of e_i cannot be determined. In fact if $\underline{e}^*$ is a solution of (1), $K\underline{e}^*$ is also a solution (where K is a scalar). For this reason any solution of (1) will be addressed as the relative flows of the network.

By assumption (VI), if one flow, say e_s, is fixed all the remaining flows are also determined. This property will be used extensively in the sequel.

To construct the steady state p.d.f. of $\underline{n}$, all we need is any non-zero solution of (1). In practice however it may be advisable for computational reasons to set the relative flow at some station at some suitable level.

As the next step we ideally take station i (i=1,...,M) out of the network and we compute the p.d.f. $\underline{P}_i(n_i)$ of the clients of station i in this condition, referred to as stand-alone.

Assume that station i is stand-alone and that the incoming flow of of client is Poisson with rate e_i determined at step 1.

The p.d.f. $\underline{P}_i(n_i)$ can be found in any queuing theory text.

Let:

$$X_i = \frac{e_i}{\mu_i} \tag{2}$$

be the relative utilization of station i. We have:

$$\underline{P}_i(n_i) = c_i x_i(n_i) \tag{3}$$

where:

$$\begin{aligned} x_i(0) &= 1 \\ x_i(n_i) &= x_i(n\text{-}1)\ \frac{X_i}{A_i(n_i)} = x_i(n\text{-}1)\ \frac{e_i}{\mu_i A(n_i)} = x_i(n\text{-}1)\ \frac{e_i}{\mu_i(n_i)} \end{aligned} \tag{4}$$

for $n_i > 0$ and

$$A_i(n_i) = \begin{cases} 1 & \text{(SS)} \\ n_i & \text{(IS)} \\ \min[n_i, L_i] & \text{(MS)} \end{cases} \tag{5}$$

L_i = number of servers

The rate at which clients are served when n_i are at station i is denoted $\mu_i(n_i)$.

Using (5) we can also write:

$$x_i(n_i) = \begin{cases} X_i^{n_i} & \text{(SS)} \\ X_i^{n_i}/n_i! & \text{(IS)} \\ X_i^{n_i}/n_i! & n_i \leq L_i \quad \text{(MS)} \\ \dfrac{X_i^{n_i}}{L_i!\, L_i^{n_i - L_i}} & n_i > L_i \end{cases} \tag{4'}$$

A problem which we have so far neglected is the possibility that some X_i/L_i might be greater than one, in which case no stationary p.d.f. for the stand alone case exists.

This case can be taken care of by simply adjusting the level of the relative flows, which does not affect the final solution $\underline{P}(\underline{n})$. Thus, using equation (3), c_i is interpreted as $\underline{P}_i(0)$.

Theorem: The steady state p.d.f. $\underline{P}(\underline{n})$ of the network is given by:

$$\underline{P}(\underline{n}) = \frac{1}{G(M,N)} \prod_{i=1}^{M} \underline{P}_i(n_i) \tag{6}$$

G(M,N) is determined using the normalization condition:

$$\Sigma\, \underline{P}(\underline{n}) = 1$$

where the sum extends to all feasible $\underline{n}$ i.e. s.t.

$$\sum_{i}^{M} n_i = N \; ; \qquad n_i \geq 0$$

The proof consists in showing that (6) satisfies the local balance equation of the network [4].

We can now justify the initial claim that any solution of (1) will work. The normalization condition (7) can be used to write:

$$G(M,N) = \sum_{\underline{n}} \prod_{i=1}^{M} \underline{P}_i(n_i)$$

where as before the sum extends to all feasible $\underline{n}$. If we multipy all relative flows e_i by a constant K, both G(M,N) and each right hand term of (8) will be multiplied by K^N, as can be seen considering expression (4'). Therefore $\underline{P}_i(n)$ will not change.

Using the same argument, it is easy to check that the value of c_i is also immaterial. Following the current literature we can therefore set:

$$c_i = 1$$

$$\underline{P}_i(n_i) = x_i(n_i) \tag{9}$$

which is the original solution found by Gordon and Newell [3].

It must be clear by now that $\underline{P}_i(n_i)$ are not the marginal p.d.f. for the stations included in the closed network. In section 2.10 however it will be shown that, for large N and a suitable choice of the relative flows, the marginal p.d.f. of the closed network tend to $\underline{P}_i(n_i)$.

2.4 Computation of G(M,N)

Using (8) and (9) we can write:

$$G(M,N) = \sum_{\underline{n}} \prod_{i=1}^{M} x_i(n_i) \tag{10}$$

where the sum is performed over all $\binom{N+M-1}{M-1}$ feasible states.

The computation of G(M,N) has been in the past performed by direct summation of all single terms in (10). Much effort has been spent to improve over such method, which is extremely time consuming. First Moore [11] derived a closed form expression for G(M,N) for the (SS) case. Later Buzen [12] suggested what is now by far the best approach to compute G(M,N).

Let us define:

$$G(m,n) = \sum_{\underline{n}} \prod_{i=1}^{m} x_i(n_i) \qquad 1 \leq m \leq M, \quad 0 \leq n \leq N \tag{11}$$

where the sum is taken over all:

$$\underline{n} = \{n_1, \ldots, n_m, 0,0,0\}$$

such that

$$\sum_{i=1}^{m} n_i = n$$

Then (10) can be rewritten in the following form:

$$G(M,N) = \sum_{n_M=0}^{N} x_M(n_M) G(M-1, N-n_M)$$

and more generally,

$$G(m,n) = \sum_{n_m=0}^{n} x_m(n_m)\, G(m-1,n-n_m) \qquad (12)$$

with the initial conditions:

$$G(0,n) = \begin{cases} 1 & \text{if } n=0 \\ 0 & \text{otherwise} \end{cases} \qquad (12')$$

Notice that:

$$G(1,n) = x_1(n_1) \qquad n_1 = 1,\ldots, N$$

$$G(m,0) = 1 \qquad m = 1,\ldots, M$$

In plain words, G(M,N) is computed by computing G(1,n) (n=1,...,N) first, using G(1,n) in (12) to compute G(2,n) and so on. If $x_i(n)$ are generated in the course of the computation, the algorithm requires 2MN (N+1) arithmetic operations [12]. Some improvement can be achieved by using Horner's rule [13].

A substantial simplication can be achieved for the (SS) case. It is trivial to show that

$$G(m,n) = G(m-1,n) + X_m G(m,n-1)$$

Other simplifications can be obtained if there are more than one IS station in the network (see next section).

The storage requirement is N+1 for the SS case, since at each stage we can store G(m,n) in the cell previously occupied by G(m-1,n), which is not required anymore. The same is true for the general case (equation (12)) if we start computing G(m,N) first, then G(m,N-1) and so on. G(m,N) can be stored in the cell previously occupied by G(m-1,N) which is not required to compute G(m,n) with n<N and so on.

2.5 The Generating Function Approach and the Closed Form Expression Of G(M,N)

The generating function approach, originally devised to compute G(M,N) is useful for deriving proofs in a compact and systematic way [14].

Define for station i the following polynomial generating function:

$$g_i(t) = \sum_{n=0}^{\infty} x_i(n)t^n \tag{14}$$

Then G(M,N) is the coefficient of the n-th lower term of the network generating function g(t):

(15)

$$g(t) = \prod_{i=1}^{M} g_i(t) = (1 + x_1(1)t + x_1(2)t^2 + \ldots)\ldots(1+x_M(1)t\ldots)$$

$$= \sum_{N=0}^{\infty} G(M,N)t^N$$

since this coefficient will contain all distinct products:

$$x_1(n_1)x_2(n_2)\ldots x_M(n_M)$$

such that:

$$\sum_{i=1}^{M} n_i = N$$

Let us derive a few explicit expression using this approach.

a.) <u>All Stations in the network are SS</u>

Assuming t is small in equation (14):

$$g_i(t) = \frac{1}{1-X_i t}$$

and

$$g(t) = \prod_{i=1}^{M} g_i(t) = \prod_{i=1}^{M} \frac{1}{1-X_i t}$$

The latter expression can be expanded to achieve:

$$g(t) = \sum_{i=1}^{M} \frac{A_i}{1-X_i t} \tag{16}$$

where:

$$A_i = \prod_{j \neq i} (1-X_j/X_i)^{-1}$$

Expanding each term in (16) we get

$$g(t) = \sum_{n=0}^{\infty} \left(\sum_{i=1}^{M} A_i X_i^n \right) t^n$$

which implies:

$$G(M,N) = \sum_{i=1}^{M} A_i X_i^N \tag{17}$$

So far we have assumed that all x_i are distinct. These expressions can be easily generalized. Suppose:

$$X_{M-1} = X_M$$

then as before:

$$g(t) = \prod_{i=1}^{M} g_i(t) = \prod_{i=1}^{M} \frac{1}{1-X_i t}$$

Expanding the expression we obtain:

$$g(t) = \sum_{i=1}^{M-2} \frac{A_i}{1-X_i t} + \frac{A_{M-1}}{1-Y_M t} + \frac{A_M t}{(1-X_m t)^2}$$

where the last two terms correspond to the double root of g(t). The last term can be written as:

$$\frac{A_M t}{(1-X_M t)^2} = \frac{A_M t}{x_M} \frac{d}{dt} \left[\frac{1}{1-X_M t} \right] = \frac{A_M t}{X_M} \frac{d}{dt} \sum_{j=0}^{\infty} (X_M t)^j$$

$$= A_M t \sum_{j=1}^{\infty} j (X_M t)^{j-1} = A_M \sum_{j=1}^{\infty} j X_M^{j-1} t^j$$

Thus, we obtain:

$$G(M,N) = \sum_{i=1}^{M-1} A_i X_i^N + A_M N X_M^{N-1}$$

b.) M'Stations are SS and the remaining IS

We assume that the stations are numbered so that the first M' are SS.

For each IS station we have:

$$g_K(t) = \sum_{i=0}^{\infty} \frac{X_K t}{i!}^i = e^{X_K t}$$

therefore:

$$\prod_{i=M'+1}^{M} g_i(t) = e^{\sum_{i=M'+1}^{M} X_i t}$$

and:

$$g(t) = e^{\Sigma X_i t} \prod_{i=1}^{M'} \frac{1}{1-X_i t} \qquad (18)$$

The result thus obtained is that when computing G(M,N) all IS stations can be substituted with one IS station with $X^* = \sum_{i=M'+1}^{M} X_i$.

A closed form expression for G(M,N) can also be derived fairly easily, but it is of little interest

It may be worthwhile to point out that the generating function approach leads exactly to the same convolutional expressions found by Buzen. In fact, let:

$$\underline{X}^i = \{x_i(0), \ldots, x_i(N)\}$$

It is immediately seen that Buzen's formula can be obtained as:

$$G(M,n) = \underline{X}^1 * \underline{X}^2 * -- * \underline{X}^M$$

which is, equivalent to (15) in the n-domain.

2.6 Marginal p.d.f.

We are going to discuss the computation of the major performance measures of the network, namely, (i) the throughput, (ii) the turnaround time and (iii) the average queue lengths.

As a preliminary step in this section we derive the closed network marginal p.d.f. $\underline{p}_i(n_i)$ of the number of clients at station i.

This p.d.f. has nothing to do with $\underline{P}_i(n_i)$ introduced in section 2.3 which refers to the stand-alone condition.

The p.d.f. $p_i(n_i)$ can be simply obtained from

$$p_i(n_i) = \frac{x_i(n_i)}{G(M,N)} \sum_{\underline{n}} \prod_{j\neq i} x_j(n_j) = x_i(n_i) \frac{G^i(M-1,N-n_i)}{G(M,N)} \tag{19}$$

where the summation is taken over all $\underline{n}$ such that:

$$\sum_{j\neq i} n_j = N-n_i$$

Function $G^i(M-1,n)$ is easily recognized as closely related to G(m,n). In fact we have:

$$G^i(M-1,n) = \underline{X}^1 * \text{ -- } \underline{X}^{i-1} * \underline{X}^{i+1} \text{ -- } * \underline{X}^M$$

If i=M, then:

$$G^M(M-1,n) = G(M-1,n)$$

The actual computation of $G^i(M-1,n)$ may be necessary for the MS station and will be discussed in section 2.8.

In view of future applications, we notice that since:

$$\sum_{n_i=1}^{N} p_i(n_i) = 1$$

from (19) we obtain:

$$G(M,N) = \sum_{n_i=1}^{N} x_i(n_i) G^i(M-1,N-n_i) \tag{20}$$

This equation generalizes equation (12).

2.7 Throughput and Utilization

The throughput T_i at station i is equal the average

number of completed clients per unit time.

We have:

$$T_i = \sum_{n_i=1}^{N} p_i(n_i)\mu_i(n_i) = \sum_{n_i=1}^{N} x_i(n_i)\mu_i(n_i) \frac{G^i_{M-1}(M-1,N-n_i)}{G(M,N)} \qquad (21)$$

Using (4) we have:

$$x_i(n_i)\mu_i(n_i) = x_i(n-1)e_i$$

Substituting in (21) and adjusting the indices:

$$T_i = e_i \sum_{n_i=0}^{N-1} \frac{x_i(n_i)G^i(M-1,N-1-n_i)}{G(M,N)} = e_i \frac{G(M,N-1)}{G(M,N)} \qquad (22)$$

The ratio of T_i/e_i is obtained using the last two components of the vector G(M,n) and does not depend on i. This was to be expected since both e_i and T_i(i=1,...,M) satisfy the flow balance equations (1).

In any network which models a flexible manufacturing system there is a station which corresponds to the Load/Unload activity. If e_L is the relative flow at this station the throughput of the overall systems is given by:

$$T_L = e_L \frac{G(M,N-1)}{G(M,N)} \qquad (23)$$

This is important because the throughput of the load/unload station is the production rate of the system.

A closely related performance is the station utilization v_i defined as the fraction of time the station is busy. We have:

$$v_i = 1 - p_i(0) = 1 - x_i(0)G^i \frac{(M-1,N)}{G(N)} = 1 - \frac{G^i(M-1,N)}{G(M)}$$

For the SS case we have:

$$T_i = \mu_i v_i$$

2.8 Turnaround

The turnaround in our model of a flexible manufacturing system is just the average overall production time of a workpiece measured from the beginning of the loading, until the completion of unloading.

Turnaround, throughput and N are related by Little's formula [15]. We have:

$$\text{turnaround} = \frac{N}{T_L} = \frac{NG(M,N-1)}{e_L G(M,N)} \tag{24}$$

where we have used (23).

2.9 Average Queue Lengths

The average queue length is defined as the average number of clients at station i (either waiting or being served):

$$Q_i = \sum_{n_i=1}^{N} n_i p_i(n_i) \tag{25}$$

The computation of Q_i will be carried out differently according the specific case.

(i) IS

If a station is IS, Little's formula can be used to derive Q_i very easily. We have

$$Q_i = \frac{T_i}{\mu_i} = X_i \frac{G(M,N-1)}{G(M,N)} \tag{26}$$

(ii) SS

Following Buzen [5] we derive Q_i using the following expression:

$$Q_i = \sum_{j=1}^{N} \text{Prob}\{n_i \geq j\} \tag{27}$$

We have:

$$\text{Prob}\{n_i \geq j\} = \sum_{K=j}^{N} X_i^K \frac{G^i(M-1,N-K)}{G(M,N)} = X_i^j \sum_{K=0}^{N-j} X_i^K(K) \frac{G^i(N-j-K)}{G(M,N)}$$

$$= X_i^j \frac{G(M,N-j)}{G(M,N)} \tag{28}$$

Substituting in (27) we obtain:

$$Q_i = \sum_{j=1}^{N} X_i^j \frac{G(M,N-j)}{G(M,N)} = X_i \frac{G(M+1,N-1)}{G(M,N)} \tag{29}$$

where G(M+1,N-1) can be computed using (13):

$$G(M+1,n) = G(M,n) + X_i G(M+1,n-1) \qquad n=1,\ldots,N-1$$

(iii) MS

There are important instances in which there is only one MS station in the network: For instance when we model the transportation system as an MS station and all the remaining station are SS.

In this special case if we know Q_i for the other stations we can obtain the required quantity noting that the Q_i must add to N.

In any other case we will compute $p_i(n_i)$ for the MS station and then use (25) to derive Q_i.

This requires the knowledge of $G^i(M-1,n)$. Again, if we assign the index M to an MS station, we know that G(M-1,n) is equal to $G^M(M-1,n)$ which will save some computations.

If there are more than 2 MS stations in the network, we definitely have to computer $G^i(M-1,n)$. A possibility indicated by Buzen [12] is to assign index M to station i, run the algorithm from the beginning and store G(M-1,n).

A better solution has been suggested in [16]. We must have:

$$G(M,n) = \sum_{n_i=0}^{n} x_i(n_i) G^i(M-1,N-n_i) \qquad n=1,\ldots,N$$

which is a set of N equations in the unknown $G^i(M-1,n)$.

Solving this system is easy if we start from n=1. We have:

$$G(M,1) = x_i(1)G^i(M-1,0) + x_i(0)G^i(M-1,1)$$

from which we derive:

$$G^i(M-1,1) = G(M,1) - x(1)$$

Using this result we obtain from the equation for n=2:

$$G^i(M-1,2) = G(M,2)-G^i(M-1,1)x_i(1)-x_i(2)$$

and in general:

$$G^i(M-1,n) = G(M,n) - \sum_{j=1}^{n} x_i(j)G^i(M-1,n-j) \qquad (30)$$

Finally we compute:

$$Q_i = \sum_{n_i=1}^{N} n_i x_i(n_i)G^i(M-1,N-n_i)$$

The computation of Q_i is not only essential to evaluate the performance of the system but plays an important role in the optimization problem, as will be shown later.

2.10 Asymptotic Behavior For Large N

In this section we discuss the behavior of the network when N becomes large. It will be shown that as N increase the throughput reaches a saturation point. A station in the network reaches its maximum service rate or capacity and acts as an exponential generator for the rest of the network which tends to behave as an open network.

We observe first that as N increases, T_i i=1,...,M cannot decrease. This may not be the case in a real system where storage is finite, but our model assumes infinite storage.

On the other hand throughput at any station cannot exceed the maximum service rate at that station. Assume that N is large enough

that a station s exists such that

$$T_s \simeq L_s \mu_s$$

Then since T_i must satisfy the flow balance equations (1) it is clear that no further increase of N contributes to the throughput of any station in the network. We say that s acts as a bottleneck for the network. Any network which models a real system has at least one bottleneck s, which is a station (SS or MS) such that:

$$\frac{X_s}{L_s} = \frac{e_s}{\mu_s L_s} = \max_i \frac{X_i}{L_i} \tag{31}$$

This result will be used to understand the asymptotic behaviour of G(M,N).

We must have:

$$\lim_{N\to\infty} T_S = \lim_{N\to\infty} e_s \frac{G(M,N-1)}{G(M,N)} = L_s \mu_s$$

or

$$\lim_{N\to\infty} \frac{G(M,N-1)}{G(M,N)} = \frac{L_s \mu_s}{e_s}$$

In simple words, as N increases, G(M,N) approximately obeys the linear relationship:

$$G(M,N) \approx \frac{e_s}{L_s \mu_s} G(M,N-1) \tag{32}$$

and increases, decreases or remains constant as N increases depending on whether $\frac{e_s}{L_s \mu_s}$ is greater, less than or equal to one.

This result is useful in two ways. First it points out a way to avoid computational problems due to quantities being either too large or too small. This can be achieved by solving the flow balance equations (1) and setting

$$e_s = L_s \mu_s$$

The relative flows corresponding to this solution will be called saturation flows.

The second use of the result is to prove the following

Theorem. As $N \to \infty$ the marginal p.d.f. $p_i(n_i)$ at all stations other than the bottleneck converge to the stand-alone p.d.f. $P_i(n_i)$ where the incoming flow is the saturation flow at station i.

Proof. Let us rewrite equation (19):

$$p_i(n_i) = x_i(n_i) \frac{G^i(M-1,N-n_i)}{G(M,N)}$$

The theorem will be proved by examining the asymptotic behavior of $G^i(M-1,N-n_i)$ and $G(M,N)$ for $N \to \infty$. Without loss of generality let us choose the saturation flows as the relative flows in the network. We know the that $G(M,N) \to K$ (constant) for $N \to \infty$.

We show now that the same is true for $G^i(M-1,N)$ i.e. $G^i(M-1,N) \to H$ (constant) for $N \to \infty$. This will be done by suggesting a physical interpretation for $G^i(M-1,N)$.

Assume that in the original network clients at station i are served in an infinitely short time. We say that station i has been short circuited. The relative flows for the original station are still a solution for this modified network. Since station i has zero service time, the i-th component of the state vector is always zero:

$$n_i = 0$$

and for this network:

$$G(M,N) = \underline{X}^1 * \ldots * \underline{X}^{i-1} * \underline{I} * \underline{X}^{i+1} * \ldots * \underline{X}^M \qquad (33)$$

where:

$$\underline{I} = \{1,0,\ldots 0\}$$

But (33) defines $G^i(M-1,N)$, therefore the latter can be interpreted as $G(M,N)$ for the network in which station i is shortcircuited. Since we have assumed that station i is not the bottleneck, it follows that the modified network has the same bottleneck as the original and the saturation flows of the original network are saturation flows also in the modified network. Therefore we have:

$$G^i(M-1,N) \to H \text{ (constant)} \qquad \text{for } N\to\infty \tag{34}$$

Using this result we can write:

$$p_i(n_i) = x_i(n_i)\,\frac{G^i(M-1,N-n_i)}{G(M,N)} \simeq x_i(n_i)\,\frac{G^i(M-1,N)}{G(M,N)}$$

Moreover we have by (19)

$$\frac{G^i(M-1,N)}{G(M,N)} = p^i(0)$$

Thus for $N\to\infty$:

$$p_i(n_i) \simeq x_i(n_i)p^i(0) = P_i(n_i)$$

This theorem was first proved by Gordon and Newell for the SS case [3] and later by Muntz [17].

An assumption made by the multicommodity flow model [Vol. II] is that each station may be modelled as a stand-alone queue. The theorem shows that this assumption is consistent with the network of queues approach provided N is large.

3. FLOW OPTIMIZATION IN A NETWORK OF QUEUES

3.1 Introduction

The flow optimization problem discussed in the section is also referred to as the scheduling of loading in a flexible manufacturing plant [6,7]. It is a control activity which decides off-line which paths the workpieces will follow within the system and which operations will be carried out at each station in the path.

The usual practice is to assign operations in such a way as to have an equal workload at each machine or at any rate to avoid bottlenecks. This approach does not take into consideration other components of the production time, namely the waiting time associated with the queues which build up at the stations.

The next sections will make use of the model just discussed to state an optimization problem which takes into account the queueing times, based on the simple idea that to each set of flows there corresponds a set of relative utilizations X_i which determine uniquely the throughput at the unloading station, given a number N of workpieces in the system. Section 3.2 will state the approach as a nonlinear optimization problem. In Section 3.3 we discuss some algorithms to compute the gradient of the objective function and finally in Section 3.4 we briefly review the properties of the problem.

3.2 Statement of the Problem

Assume we have a flexible manufacturing system modelled as a network of queues of the class just discussed. The system's task is to manufacture a mix of R different parts in the minimum time, given a fixed number of pallets N. The part mix is defined by the constants $K_r (r=1,\ldots,R)$ which are the specified ratios of the production of part r to total production.

A total of $S_r (r=1,\ldots,R)$ different strategies are available to manufacture part r. The strategy specifies the sequence of stations by part r and the processing times at each visited station. All strategies are assumed to be specified in advance. The optimization problem in this simplified version amounts to finding the fraction λ_{rs} of total production

devoted to manufacturing part r according to strategy s(s=1,...,S_r). Fractions λ_{rs} are chosen to maximize total production or equivalently to minimize the production time or turnaround (sect. 2.8).

Without loss of generality we can assign label M to the unloading station and set $e_M = 1$.

By definition:

$$\sum_{r=1}^{R} K_r = 1 = e_M \tag{1}$$

Because we set $e_M=1$, we obtain:

$$\sum_{s=1}^{S_r} \lambda_{rs} = K_r \qquad \sum_{r=1}^{R} \sum_{s=1}^{S_r} \lambda_{rs} = 1 = e_M \tag{2}$$

Let us first derive the relative flows e_i in term of λ_{rs}. To do so we must know the total number of visits $V^i(r,s)$ made to station i by workpiece r under strategy s before it reaches the unloading station. Specifically $V^i(r,s)$ may be zero if station i is not required, one or more if the station is required at least once. This information is available since all strategies are known.

Then we obviously have:

$$e_i = \sum_{r=1}^{R} \sum_{s=1}^{S_r} \lambda_{rs} V^i(r,s) \tag{3}$$

We are now going to derive the relative utilizations X_i as a function of λ_{rs}.

The easiest way to do this is to remember that X_i is just the amount of work performed at station i in the time unit:

$$X_i = e_i \frac{1}{\mu_i} = e_i \tau_i$$

τ_i = average processing time.

Let $W^i(r,s)$ be the total expected working time required by a single workpiece r at station i under strategy s before it reaches the unloading terminal. $W^i(r,s)$ may be zero if under strategy s part r does not need station i. If greater than zero, it corresponds to the overall time spent by the workpiece during the one or more visits at the

station. This data is known since strategies are all known.

Then the overall working time spent per unit time by station i to produce part r under strategy s is equal to $W^i(r,s)\lambda_{rs}$. Adding we have:

$$X_i = \sum_{r=1}^{R} \sum_{s=1}^{S_r} \lambda_{rs} W^i(r,s) \qquad (4)$$

We are now able to evaluate the performance of the system for a given set of flows λ_{rs}. Using equations (3) and (4) and the information $w^i(R,S)$, $V^i(r,s)$ we can compute e_i and X_i and from there derive both throughput and turnaround.

We can in fact look for the best set $\underline{\lambda}$:

$$\underline{\lambda} = \{\lambda_{ps} \mid r=1,\ldots,R;\ s=1,\ldots S_p\}$$

which is the solution of the following optimization problem:

$$\min\ t(\underline{\lambda}) = -\log \frac{G(M,N-1 \mid \underline{X}(\underline{\lambda}))}{G(M,N \mid \underline{X}(\underline{\lambda}))}$$

over $\underline{\lambda}$

s.t.

$$\sum_{s=1}^{S_r} \lambda_{rs} = K_r \qquad r=1,\ldots,R \qquad (5)$$

$$\lambda_{rs} \geq 0 \qquad r=1,\ldots,R;\ s=1,\ldots,S_r$$

The objective function is just the logarithm of the turnaround (eq.2-24), if we disregard a constant term. The logarithm has been introduced because it simplifies the computation of the gradient. The notation stresses the dependence of $G(M,\cdot)$ on the choice of $\underline{\lambda}$, by means of equation (4).

3.3 Features of the Problem

It is conjectured that function $t=G(M,N-1 \mid \underline{X}(\lambda))/G(M,N \mid \underline{X}(\lambda))$ is concave over $\underline{X}$. Several tests support this conjecture, but a formal

proof has not yet been derived. As Log(·) is increasing and concave, it follows that the objective function is convex.

Equations (2) and (4) define a convex set of $\underline{X}$ and therefore the problem is a convex program. Any local minimum is a global minimum and Kuhn-Tucker conditions are both necessary and sufficient.

Let us write the weak Lagrangian of this problem:

$$L = t(\underline{\lambda}) - \sum_{r=1}^{R} \theta_r \left(\sum_{s=1}^{S_r} \lambda_{rs} - K_r \right) - \sum_{r=1}^{R} \sum_{s=1}^{S_r} \phi_{rs} \lambda_{rs}$$

The Kuhn-Tucker conditions are

$$\nabla_{\lambda} L = 0$$

$$\phi_{rs} \lambda_{rs} = 0 \tag{6}$$

$$\phi_{rs} \geq 0 \qquad (r=1,\ldots,R; \quad s=1,\ldots,S_r)$$

Expanding the gradient we find that a set of flows is optimal if for each r a θ_r exists such that:

$$\frac{\partial t(\lambda)}{\partial \lambda_{rs}} = \theta_r \qquad \text{if} \quad \lambda_{rs} > 0$$

$$\frac{\partial t(\lambda)}{\partial \lambda_{rs}} \geq \theta_r \qquad \text{if} \quad \lambda_{rs} = 0$$

3.4 Computation of the Gradient of $t(\underline{\lambda})$

In this section we discuss the algorithm to compute the gradient of the objective function $t(\underline{\lambda})$. Using equations (4) and the chain rule:

$$\frac{\partial t(\lambda)}{\partial \lambda_{rs}} = \sum_{i=1}^{M} \frac{\partial t(\underline{\lambda})}{\partial X_i} \frac{\partial X_i}{\partial \lambda_{rp}} = \sum_{i=1}^{M} \frac{\partial t(\underline{\lambda})}{\partial X_i} \, w^i(r,s) \tag{7}$$

Moreover we have:

$$\frac{\partial t(\underline{\lambda})}{\partial X_i} = \frac{1}{G(M,N)} \frac{\partial G(M,N)}{\partial X_i} - \frac{1}{G(M,N-1)} \frac{\partial G(M,N-1)}{\partial X_i} \tag{8}$$

It remains to compute the derivative $\frac{\partial G(M,N)}{\partial X_i}$. We rewrite equation (2-20'):

$$G(M,N) = \sum_{n_i=0}^{N} x_i(n_i) G^i(M-1,N-n_i)$$

Remembering that:

$$x_i(n_i) = \frac{X_i^{n_i}}{A_i(n_i)}$$

we obtain:

$$\frac{\partial G(M,N)}{\partial X_i} = \sum_{n_i=1}^{N} n_i \frac{X_i^{n_i-1}}{A_i(n_i)} G^i(M-1,N-n_i) =$$

$$= \frac{1}{X_i} \sum_{n_i=1}^{N} n_i \frac{X_i^{n_i}}{A_i(n_i)} G^i(M-1,N-n_i) \tag{9}$$

Dividing both sides of (9) by G(M,N) we find:

$$\frac{1}{G(M,N)} \frac{\partial G(M,N)}{\partial X_i} = \frac{1}{X_i} \sum_{n_i=1}^{N} n_i x_i(n_i) \frac{G^i(M-1,N-n_i)}{G(M,N)} = \frac{1}{X_i} Q_i(N) \tag{10}$$

where in the last expression we have used equation (2-19) and (2-25). Equation (10) extends a result found by Bhandiwat and Williams for the SS case [14].

Substituting (10) in (8) we find:

$$\frac{\partial t(\underline{\lambda})}{\partial X_i} = \frac{Q_i(N) - Q_i(N-1)}{X_i} \tag{11}$$

what this striking expression says is that to evaluate the impact of an increase of the relative utilization upon the turnaround we must evaluate the queues at the station for N and N-1 clients in the system. Section 2.9 has already dealt with this problem.

4. THE ASYMPTOTIC FLOW OPTIMIZATION PROBLEM

4.1 Introduction

In this section we study the flow optimization problem as N becomes large or equivalently when a saturation condition is approached. This condition is interesting because real life systems tend to be operated in close to saturation mode. If the production requirements drop, for instance, the system will be operated over two shifts in saturated condition rather than over three unsaturated shifts. Failures which do not stop the plant altogether also tend to lead to a saturation condition.

It turns out that the problem is a linear program (LP) and therefore relatively easy to solve. Solutions to this special problem may be used in the general case as a first guess solution.

4.2 The Optimization Problem for Large N

The objective function of problem (5) is the average turnaround, disregarding the logarithm. For convenience we will state an equivalent problem where we attempt to maximize the average throughput of the system:

$$\max T_m(\underline{\lambda}) = \frac{G(M,N-1|\underline{X}(\underline{\lambda}))}{G(M,N|\underline{X}(\underline{\lambda}))} \tag{12}$$

over $\underline{\lambda}$ s.t.

$$\sum_{s=1}^{S_r} \lambda_{rs} = K_r \qquad r=1,\ldots,R$$

$$\lambda_{rs} \geq 0 \qquad r=1,\ldots,R;\ s=1,\ldots,S_r$$

We have assumed that the unload station has index M and that $e_M=1$.

Problem (12) is equivalent to problem (5) as any solution of (5) is a solution of (12), throughput and turnaround being related by Little's formula.

In Section 2.10 we showed that as N grows we have approximately:

$$\frac{G(M,N-1)}{G(M,N)} \simeq \frac{L_s \mu_s}{e_s} = \frac{L_s}{X_s}$$

where:

$$\frac{L_s}{X_s} = \min_i \frac{L_i}{X_i} \qquad i=1,\ldots,M \tag{13}$$

and excluding the IS stations. As N grows, problem (12) can be stated in the following asymptotic form:

$$\max T_M(\underline{\lambda}) = \min_i \frac{L_i}{X_i(\underline{\lambda})}$$

over $\underline{\lambda}$ s.t.

$$\sum_{s=1}^{S_r} \lambda_{rs} = K_r \qquad r=1,\ldots,R \tag{14}$$

$$\lambda_{rs} \geq 0 \qquad r=1,\ldots,R;\ s=1,\ldots,S_r$$

We now show that this problem can be stated as an LP problem.

Let:

f_{rs} = number of workpieces produced per unit time of type r under strategy s for a given set $\underline{\lambda}$

Then remembering that:

$$\sum_{r=1}^{R} \sum_{s=1}^{S_r} \lambda_{rs} = e_M = 1$$

and noticing that both λ_{rs} and f_{rs} are solutions of the flow balance equations, we have;

$$f_{rs} = T_M(\underline{\lambda})\lambda_{rs} \qquad r=1,\ldots,R;\ s=1,\ldots,S_r \tag{15}$$

and

$$T_M(\underline{\lambda}) = \sum_{r=1}^{R} \sum_{s=1}^{S_r} f_{rs} \tag{16}$$

From (13) we have:

$$T_M(\underline{\lambda}) \leq \frac{L_i}{X_i} \qquad i=1,\ldots,M$$

or:

$$T_M(\underline{\lambda})X_i \leq L_i \qquad i=1,\ldots,M \tag{17}$$

Using (4) and (15) in (17) we obtain:

$$\sum_{r=1}^{R} \sum_{s=1}^{S_r} f_{rs} W^i(r,s) \leq L_i$$

Using variables f_{rs} $(r=1,\ldots R;\ s=1,\ldots S_r)$, problem (14) can be stated equivalently in the following way:

$$\max T_M(\underline{f}) = \sum_{r=1}^{R} \sum_{s=1}^{S_r} f_{rs} \tag{18}$$

over $\underline{f}$ s.t.

$$\sum_{s=1}^{S_r} f_{rs} = K_r \sum_{r=1}^{R} \sum_{s=1}^{S_r} f_{rs} \qquad r=1,\ldots,R$$

$$\sum_{r=1}^{R} \sum_{s=1}^{S_r} f_{rs} W^i(r,s) \leq L_i \qquad i=1,\ldots,M$$

$$f_{rs} \geq 0 \qquad r=1,\ldots,R;\ s=1,\ldots,S_r$$

leading to an LP problem.

The first constraint takes into account the mix requirements and has been derived from the first constraint in problem (14) using (15). The second constraint makes sure that at no station is the work station capacity exceeded. Problems (14) and (18) are equivalent in the sense that their optimal solution lead to the same value of the objective functions. If one is interested in λ_{rs} , we can derive them from f_{rs} using this expression:

$$\lambda_{rs} = \frac{f_{rs}}{\sum_{r,s} f_{rs}}$$

5. A CASE STUDY

5.1 Introduction

A simple case study has been examined in some depth using the multi-commodity flow appraoch (MFA), the network approach (NA) and the asymptotic network approach (ANA).

The major results are the following:

(i) the three approaches lead to approximately the same solution;

(ii) at the optimal point stations are not balanced;

(iii) a parametric study, shows that ANA and NA solutions are in-good agreement even for relatively small N.

5.2 The Physical Model

The system to be modeled is shown in Fig. 2. It includes two work stations and a load/unload station. Two parts are processed, in a ratio of 2:1. Part 1 must be worked at station 1 and station 2. Part 2 requires only one operation which can be performed either at station 1 or 2. These are the exact data.

$M=3$ $R=2$ $S_1=1$ $S_2=2$ $K_1=2/3$ $K_2=1/3$ $N=30$

(r.s)	W^1	W^2	W^3
1,1	1.5	1.67	1.0
2,1	1.5	-	1.0
2.2	-	1.67	1.0

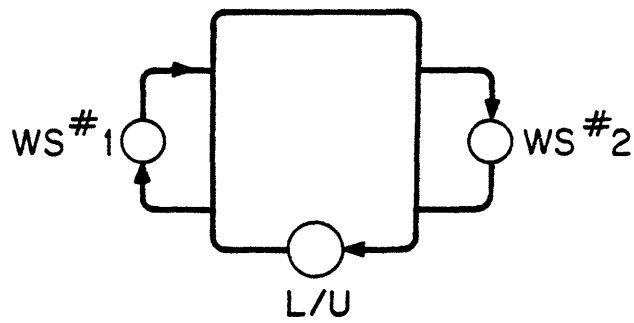
The optimal MFA solution specifies that 63% of workpiece-2 must be produced according to strategy 1. In our notation:

$$\lambda_{21} = \frac{1}{3} \cdot 0.63 = .21$$

5.3 The Network Model and the Results

The system has been modelled as a three SS station network. To simplify the problem, the transportation system has not been included in the network and therefore workpieces move from one station to the other with no delay.

A CASE STUDY



$M = 3$ $P = 2$ $K_1 = \frac{2}{3}$ $K_2 = \frac{1}{3}$ $N = 30$

$S_1 = 1$

$S_2 = 2$

(p,s)	$W^1(p,s)$	$W^2(p,s)$	$W^3(p,s)$
1.1	1.5	1.67	1.0
2.1	1.5	--	1.0
2.2	--	1.67	1.0

Fig. 2 Layout and Parameters for Case Study System

We nevertheless believe that the results of the NA and MFA problems can be compared, the major reason being that in the MFA problem the transportation system operates also at a very high speed.

The behaviour of the throughput has been studied as a function of both λ_{21} and N, entering the data manually.

Figure 3 shows the throughput for fixed N as a function of λ. The graph is nearly flat and the maximum occurs for a value which is in good agreement with the MFA solution.

In Figure 4 the throughput is plotted as a function of N, λ being a parameter. It is clear that the system saturates well before N=30 and that for small N the slopes of the curves are similar, while remarkable differences arise for higher N.

Fig. 5 shows how the optimal split varies as a function of N. When N=1 the optimal policy is to use strategy 1, that is to use the faster of the two machines; as N grows however machine 2 becomes more and more utilized. The striking result is however that the value of λ which holds for N=30 is approximately achieved at N=5, which is relatively far from the saturation.

In Figure 6 the queue lengths of the three stations are plotted for N=30, as a function of λ. It might appear that station 1 and 2 are balanced, while in fact they are not as it will be shown in the next section.

5.4 The Kuhn-Tucker Condition for the NA Problem

Using equations (3.7) and (3.11) it is easily seen that the Kuhn-Tucker conditions for this case study are:

$$X_1^{-1}(Q_1(N-1)-Q_1(N))1.5 + X_3^{-1}(Q_3(N-1)-Q_3(N))$$

$$= X_2^{-1}(Q_2(N-1)-Q_2(N))1.67 + X_3^{-1}(Q_3(N-1)-Q_3(N))$$

or alternatively:

$$\frac{1.5}{1.67}\ \frac{Q_1(N-1)-Q_1(N)}{Q_2(N-1)-Q_2(N)} = \frac{X_1}{X_2}$$

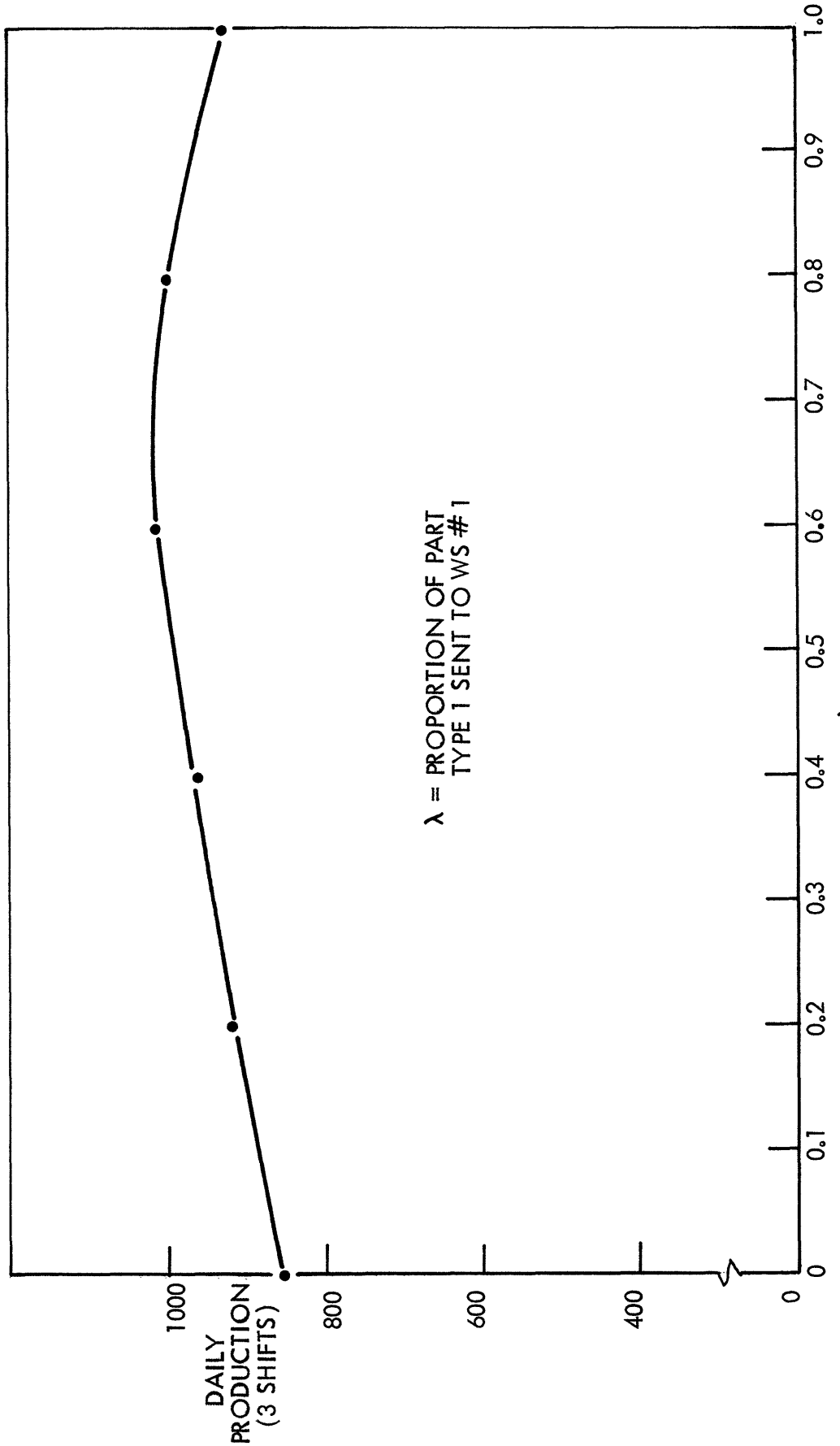


Fig. 3 Production P as a Function of λ (N = 30)

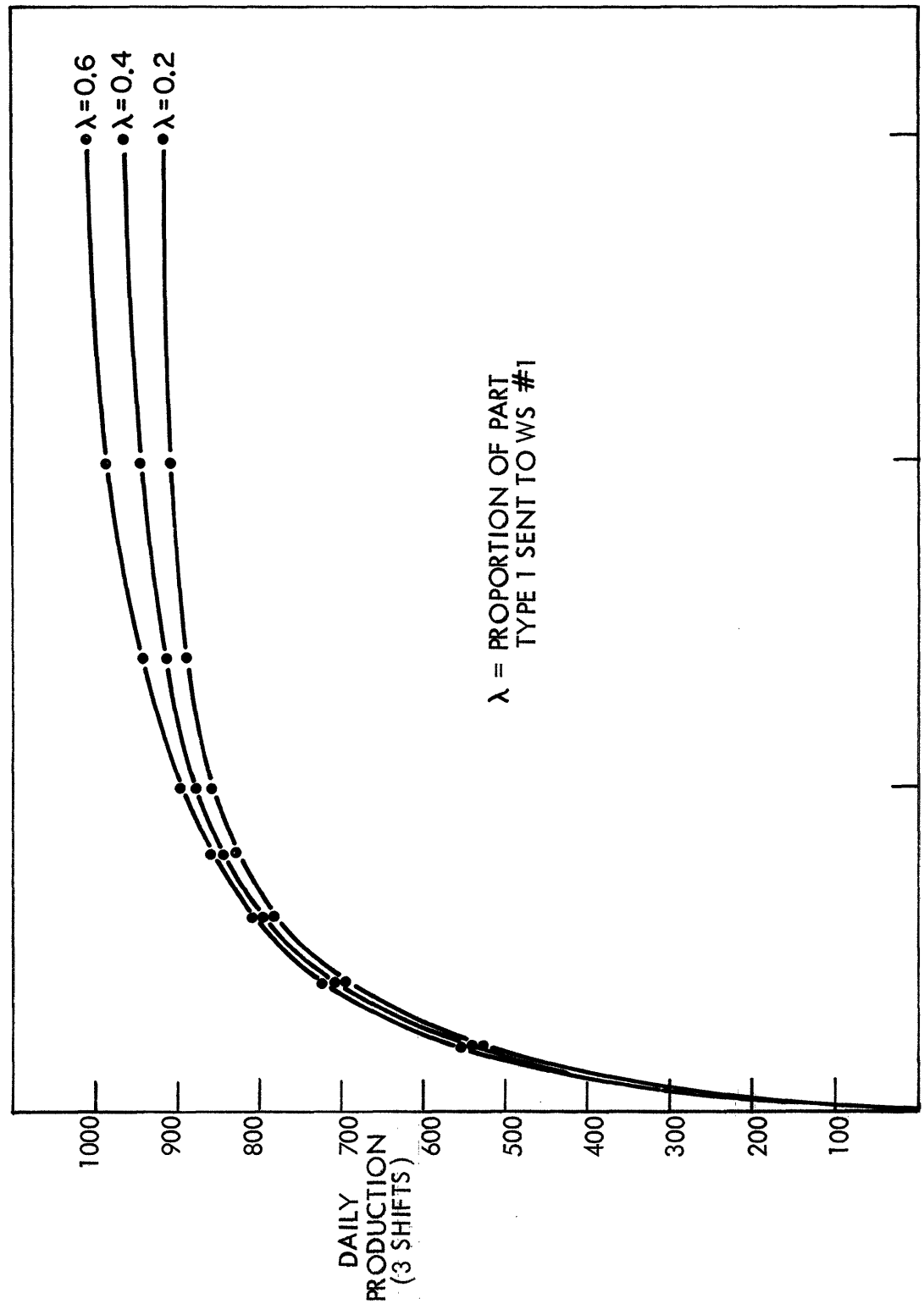


Fig. 4 Production as a Function of N for Various λ

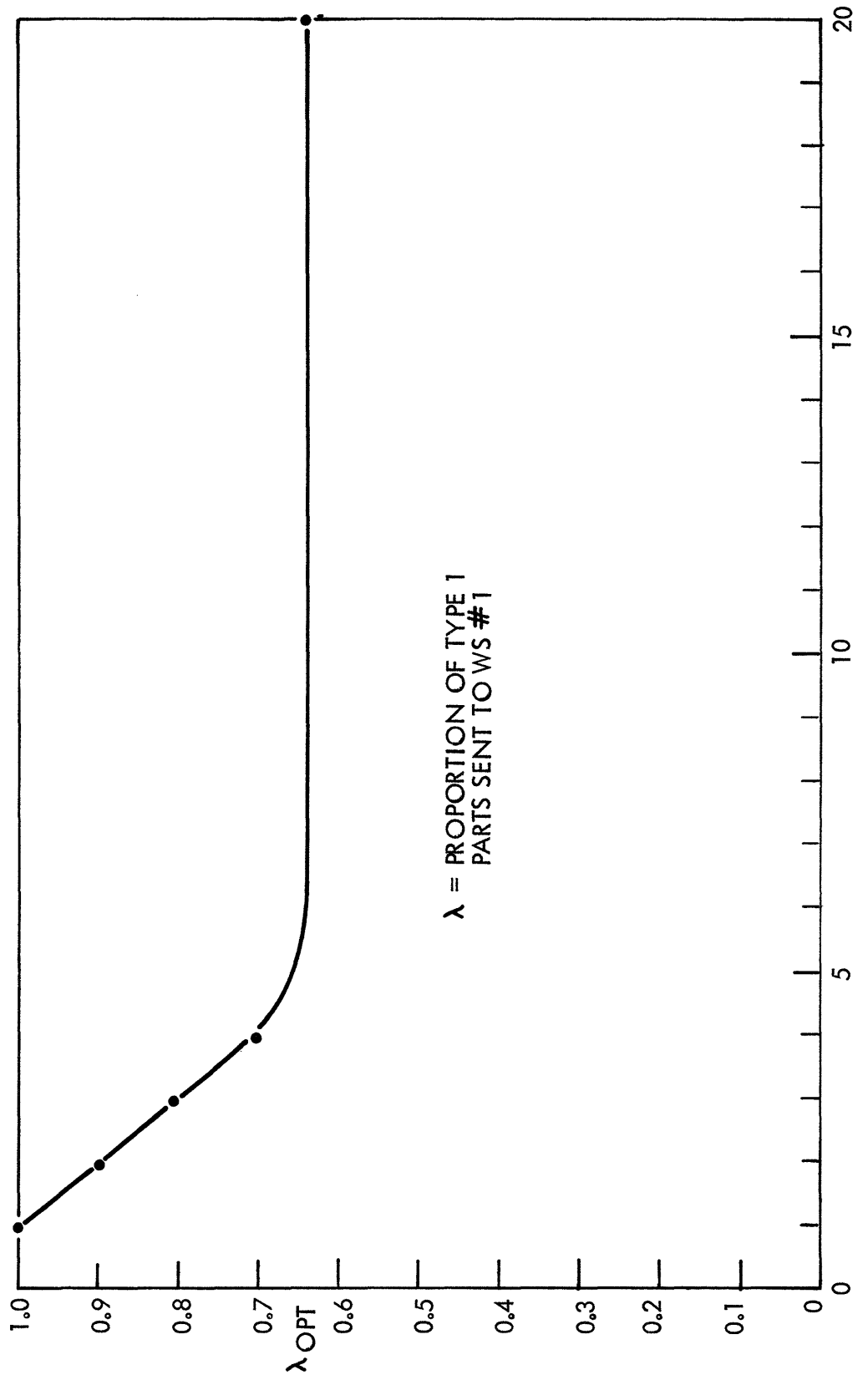


Fig. 5 Optimum Value of λ as a Function of N

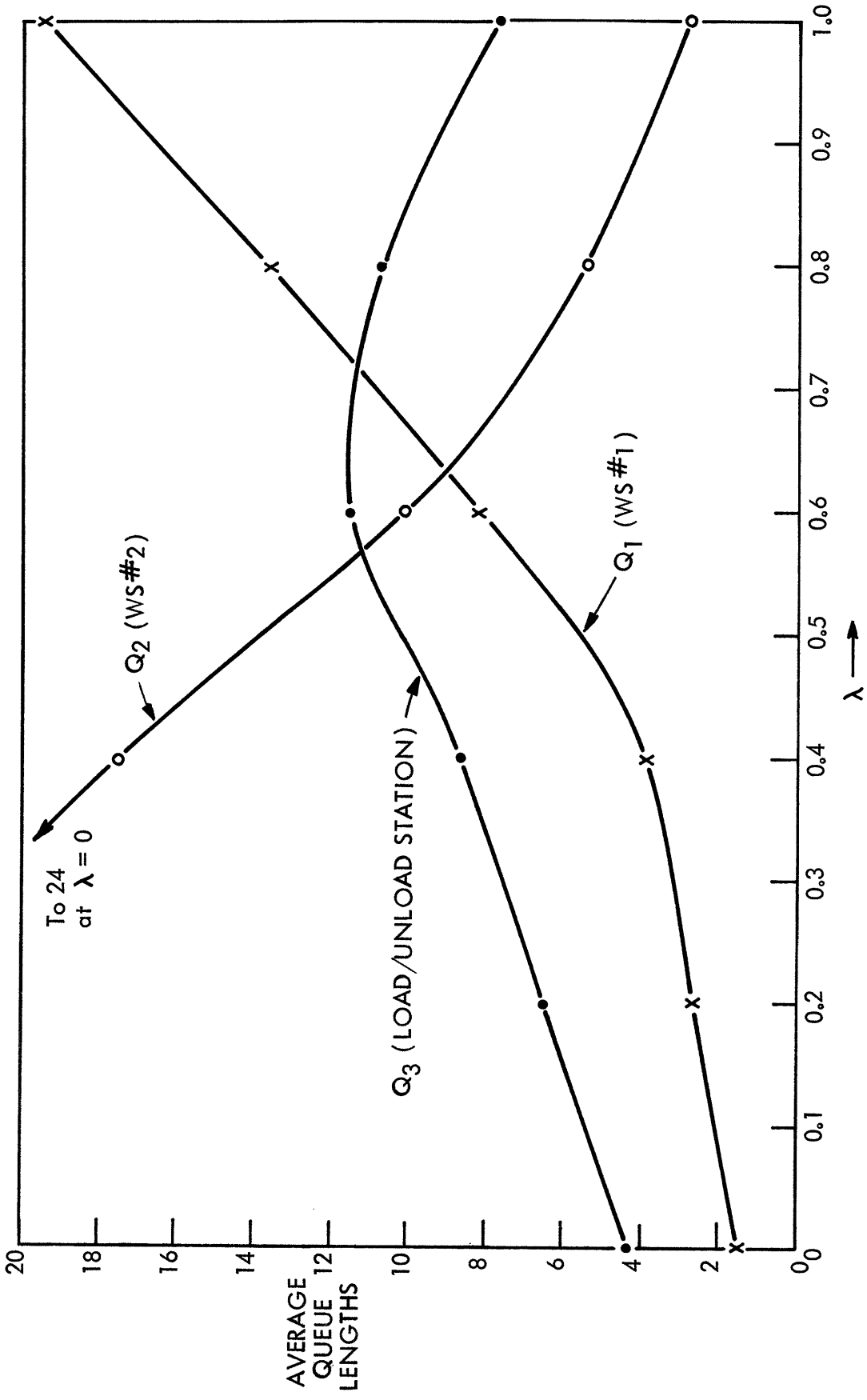


Fig. 6 Average Station Queue Lengths vs λ

This condition has been checked at the optimal point, which is $\lambda_{21} = .21$.

Notice that at the optimal point $X_1 \neq X_2$, and therefore the two stations are not balanced. In fact if $X_1 = X_2$, this would imply that $Q_1(N) = Q_2(N)$ and $Q_1(N-1) = Q_2(N-1)$ and the Kuhn-Tucker condition would be satisfied only if $w^1(2,1) = w^2(2,2)$. At the optimal point, the faster of the two station is more loaded than the other. Buzen [5] has observed a similar pattern in the computer networks.

5.5 The Asymptotic Problem

For the case on hand the problem is the stated as follows:

$$\begin{aligned} \max \quad & f_1+f_{21}+f_{22} \\ \text{s.t.} \quad & \\ & f_1 - 2(f_{21}+f_{22}) = 0 \\ & (f_1+f_{21})1.5 \leq 1 \\ & (f_1+f_{22})1.67 \leq 1 \\ & (f_1+f_{21}+f_{22})1.0 \leq 1 \end{aligned}$$

Constraints (19) - (20) reflect the fact that the utilization of the work-stations cannot exceed unity in a stable system. The problem can be simplified by using the production ratio constraint (18) to eliminate f_1:

$$\begin{aligned} \max \quad & f_{21}+f_{22} \\ \text{s.t.} \quad & \\ & 4.5f_{21} + 3f_{22} \leq 1 \\ & 3.34f_{21} + 3.0f_{22} \leq 1 \end{aligned}$$

The solution of the L.P. problem is:

$$f_{21} = 0.16 \qquad f_{22} = 0.09$$

which means that 63% of part 2 should be machined at station 1. The solution is the same as that obtained by the MFA: the required computational effect is so small that the computation has been performed manually.

6. Future Work

In this section we briefly point out areas of future investigation or activity

(i) Algorithm Implementation

A program will be developed to implement a first version of the optimization problem stated in Section 3. The first runs will tell, if this approach is feasible or if the computational effort is excessive. Even if the approach is shown to be feasible it is felt that more efforts have to be spent to optimize the algorithms to compute the network performances.

(ii) Separation Issues

Real life systems are often made of separable cells, each cell being able to perform some manufacturing operation which could not be carried out elsewhere. This property might be exploited to divide the optimization problem into smaller ones. Theorems are available to study a network at a local level using a Norton's theorem approach [20].

(iii) Strategy Generation

Since the number of strategies to be examined is very large an issue to be addressed will be the implementation of a column generation approach [21,22] for our problem. This issue is discussed in another volume of this report [23].

(iv) Validation

Even if its accuracy is excellent, the network of queues model is still an approximation to the actual behavior of a real system. In Section 2.2 we have pointed out how unrealistic some of the assumptions of the model are. It follows that an area of research will be the validation of the optimization procedure. A major objective will be to compare this approach with more traditional (and less time-consuming) algorithm to assess whether the obtainable gains justify the amount of computational effort which the algorithm requires. The validation activity will necessarily require an extensive ad detailed simulation of some real systems following the same outline of [18].

7. Conclusions

A new approach to the flow optimization problem in a flexible manufacturing system has been discussed. The approach uses a wide class of network of queues to model the performance of the system for a given distribution of flows.

The algorithms required by the optimization problem have been stated. The behavior of the network for large N has been examined and it has been found that the queue model assumed by the multicommodity flow approach is consistent with the network approach for large N.

The optimization problem has been briefly discussed and the expressions for the gradient of the objective function derived.

It has been shown that as N grows this problem degenerates into a simpler min-max problem which can be stated as an LP.

A case study has been examined. The results of it indicate that the solutions derived using the network approach are in good agreement with the solutions derived using the multicommodity flow approach. Moreover the solutions of the asymptotic problem seem to apply also for conditions far from saturation.

REFERENCES

1. J.J. Solberg, "Optimal Design and Control of Computerized Manufacturing Systems", Proceedings AIIE Systems Eng. Conf., Boston, Mass. 1976.

2. J. Ward, "Numerical Experience with a closed Network of Queues Model," Complex Materials Handling and Assembly Systems, Volume VIII, Final Report ESL-FR-834-8, Electronic Systems Laboratory M.I.T., September 1978.

3. W. Gordon, G. Newell, "Closed Queueing Systems with Exponential Servers", Oper. Res. 15, 2, Apr. 67.

4. F. Baskett, K. Chandy, R. Muntz, F. Palacios, "Open, Closed and Mixed Networks of Queues with Different Classes of Customers", J.A.C.M. 22, 2 Apr. 75.

5. J. Buzen, Queuing Networks Models of Multiprogramming", Ph.D. Thesis, Division of Engineering and Applied Science, Harvard University, Cambridge, Mass. 1971.

6. G.K. Hutchinson, "The Control of Flexible Manufacturing Systems: Required Information and Algorithm Structures in Information Control Problems in Manufacturing Technology", Y. Oshima ed. Pergamon Press, Oct. 77.

7. G.M. Secco-Suardo, "Versatile Manufacturing: a State of the Art", FIAT Res. Report, FIAT Research Center, Jan. 78 (in Italian)

8. J.J. Solberg, "Quantitative Design Tools for Computerized Manufacturing Systems" (Florida).

9. J.R. Jackson, Job-Show-Like Queuing Systems. Management Science, 10, 131-142, 1963.

10. R.R. Muntz, "Poisson Departure Processes and Queuing Networks", IBM, Research Report RC4145, Dec. 72.

11. F. Moore, "Computational Model of a Closed Queuing Network with Exponential Servers", IBM, J. Res. Devel, Nov. 1972.

12. J. Buzen, "Computational Algorithms for Closed Queuing Networks with Exponential Servers", Comm. ACM 16, n° 9, Sept. 73.

13. M. Reiser, H. Kobayashi, "Horner's Rule for the Evaluation of General Closed Queuing Networks", Comm. ACM, 18, n°10, Oct. 75.

14. A. Williams, R. Bhandiwad,"A Generating Function Approach to Queuing Network Analysis of Multiprogrammed Computers", Networks, 6, 1-22, 1976.

15. L. Kleinrock,"Queuing Systems", Vol. II, Wiley 1976.

16. G. Balbo, S. Bruell, H. Schwetman,"Computational Aspects of Closed Queuing Networks with Different Customer Classes", Purdue University Dept. Computer Science, CSD-TR210, July 1977.

17. R.R. Muntz, J. Wong,"Asymptotic Properties of Closed Queuing Network Models", Proc. 8th Ann. Princeton Conf. Information, Sciences and Systems (Mar. 74).

18. K. Stecke,"Experimental Investigation of the Scheduling Problems of a Particular Computerized Manufacturing System." Conference on Optimal Planning of Computerized Manufacturing Systems. Purdue University, Nov. 77.

19. M. Avriel,"Nonlinear Programming", Prentice Hall, 1976.

20. K. Chandy, U. Herzog, L. Woo,"Parametric Analysis of Queuing Networks", IBM, J. Res. Devel., Jan. 75.

21. L. Lasdon,"Optimization Theory for Large Systems", MacMillan, 1970.

22. J.E. Defenderfer,"Comparative Analysis of Routing Algorithms for Computer Networks", M.I.T., Rep. ESL-R-756, March 77.

23. J. Kimemia and S.B. Gershwin, "Multicommodity Network Flow Optimization in Flexible Manufacturing Systems", Complex Materials Handling and Assembly Systems, Volume II, Final Report ESL-FR-834-2, Electronic Systems Laboratory, M.I.T., September, 1978.